A Novel Method for Modeling RFID Data

Kun Bai, Shaoxi Li and Tao Liu
Dalian Maritime University, Dalian, China
Email: {tailandterry, shaoxi_li, liutao_youxiang}@126.com

Jing Tian
Liaoning Police Academy, Dalian, China
Email: tianjing11230@126.com

***Abstract*—The applications of Radio Frequency Identification (RFID) and Electronic Product Code (EPC) in supply chain management have vast potential in improving effectiveness and efficiencies in solving supply chain problems. RFID data, however, have their own unique characteristics – including aggregation, location, temporal and history-oriented – which have to be fully considered and integrated into the RFID data model which is constructed for RFID application system. In this paper, we use an expressive temporal data model for RFID data. This data model is based on Entity-Relation (ER) model with minimum extension and highlights the state history and temporal semantics of the RFID system business processes. We also show common RFID data tracking and monitoring types, and propose methods to express such queries based on this data model.**

***Index Terms*—RFID, EPC, data model, semantics**

I. INTRODUCTION

A. Background

With the great ability of identifying each object by a unique Electronic Product Code (EPC), Radio Frequency Identification (RFID) technology enables capturing large volumes of data at high speed and can be used for identifying, locating, tracking and monitoring physical objects without line of sight [1]. The potential of RFID for increasing supply chain efficiency has been stressed repeatedly by practitioners and researchers alike. The accuracy and detail of RFID data are expected to open up new and more efficient ways to steer the supply chain and reduce many of the inefficiencies plaguing today's businesses such as wrong deliveries, shrinkage and counterfeiting. Since production and distribution of physical goods are seldom in the hands of one organization and efficiency gains in supply chains oftentimes emanate from centralized supply chain control, it often makes sense to share selected RFID data among supply chain participants. Through the automatic data collection, RFID technology can achieve greater visibility and product velocity across supply chains, more efficient inventory management, easier product tracking and monitoring, reduced product counterfeiting and theft, and much reduced labor cost. RFID offers a possible alternative to bar code identification systems and it facilitates applications like item tracking and inventory management in supply chain.

RFID technology holds the promise to streamline supply chain management, facilitate routing and distribution of products, and reduce costs by improving efficiency. On the other hand, there is a chasm between the physical world and the interpreted world through reader observations. These observations need to be automatically interpreted and semantically transformed into business logic data before they can be integrated into business applications.

B. Characteristics of RFID Data

The RFID systems are being used in variety of applications. Despite of this diversity of the data generated out of the RFID applications, RFID data share some common fundamental characteristics, which have to be fully considered in RFID data application systems [2].The characteristics of RFID data are as follows:

Defective Data and Requirement of Integration: The RFID system has parallel transponders and receivers, thus information generated every second may not be reliable. Sometimes, it is mingled with other tag values or other environmental hindrances. The low level RFID observations need to be transformed and aggregated into semantically meaningful data suitable to corresponding application level.

Temporal and Dynamic: RFID applications dynamically generate observations and the data carry state changes. All reader observations are associated with the timestamps when the readings are made; objects' locations change along the time; and all EPC related transactions are also associated with time. It is essential to model all such information in an expressive data model suitable for application level interactions, including tracking, monitoring, and application integration.

Implicit Semantics: In an RFID system, a reader observation comprises of the reader EPC, the observed EPC value of an RFID tag, and the timestamp when the observation occurs. These data carry implicit information, such as changes of states and business processes (e.g., change of locations), and further derived information such as aggregations (e.g., containment relationship among objects).

Here we describe the event semantics a little more. Each event encodes several assertions which collectively

doi:10.4304/jsw.7.4.835-843

define the semantics of the event. Some of these assertions say what was true at the time the event was captured. Other assertions say what is expected to be true following the event, until invalidated by a subsequent event. These are called, respectively, the retrospective semantics and the prospective semantics of the event. For example, if Product #23 enters building #5 through door #6 at 14:33pm, then one retrospective assertion is that "Product #23 was observed at door #6 at 14:33pm", while a prospective assertion is that "Product #23 is in building #5". The key difference is that the retropective assertion assertion refers to a specific time in the past ("Product #23 was observed…"), while the prospective assertion is a statement about the present condition of the object ("Product #23 is in…"). The prospective assertion presumes that if Product #23 ever leaves building #5, another event will be recorded to supercede the prior one.

Streaming and Tremendous Amount: RFID system may generate terabytes of data in single day on an average and RFID data are generated quickly and automatically, and accumulated for tracking and monitoring. Thus to deal with such flood of data is a challenging task. Therefore, the data require a scalable storage scheme to assure efficient queries and updates.

C. Previous Data Model and Our Approach to RFID Data Modeling

RFID technology makes it possible to collect large amount of data for tracking and identifying physical objects along their history and real-time monitor physical objects and their environment for monitoring applications. So the data model of RFID application system must be efficient for some queries on tracking and monitoring objects and has semantic meanings as much as possible.

However, little research has been conducted on how to effectively model RFID data. It has been widely recognized that temporal aspects of database schemas are prevalent and difficult to model using the ER model. As a result, how to enable the ER model to properly capture time-varying information has been an active area of the database research community. H.Gregersen and C.S.Jensen [17] provided a comprehensive overview of temporal ER modeling. Their model meets a need for consolidating and providing easy access to the research in temporal ER modeling. The focus of their work was the examination of how the extensions of the ER model into temporal ER models are shaped. Although they defined different kinds of properties for temporally enhanced ER models, their work seldom involved in implicit semantics of the RFID data.

Fusheng Wang et al. proposed a data model for Siemens RFID Middleware [2]. Their model includes four primary static entities: objects, sensors/readers, locations, and transactions. These entities interact with each other to generate state changes (location change, containment change) and event changes (observations and transaction items). These changes are modeled as dynamic binary relationships which associate with timestamps (event changes) or time intervals [tstart, tend]. Their dynamic relationship ER model is simple but the tables created according to the dynamic relationship ER model and the fields in the corresponding tables are not efficent for describing the transaction information such as business steps and dispositions of objects.

Mark Harrison summarized RFID data characteristics and provided some reference relations to model the data [3]. Their paper examines the different fundamental categories of data which one might want to access via an EPC Information Service (EPCIS), then proposes three fundamental categories of simple query, which allow more complex queries to be broken down into tractable simple queries which an EPCIS could answer. In their model, RFID data are modeled as events. Thus, RFID state history and temporal semantics of business processes are implicit. The timestamped data in their model includes observation, transaction, containment, and measurement. Each type is represented in a three-dimensional space which has a timestamped axis and other two axes representing two entities joining that dynamic relation. Collected temporal data are viewed as points in the corresponding 3D spaces. The simple temporal query in the space is the process of looking for appropriate plane, line, or point. The complex temporal query is decomposed into numerous simple queries whose results are joined and filtered later. However, this model is not efficient enough for complex queries and queries related to business context because obviously, three dimensions are just enough for describing the minimum information about a dynamic relationship.

Sun RFID software [4] models each static entity type in three different tables: the first for the entity type itself including identifer and the attributes that all entities share in common, the second for specific attributes intended for each entity, the third for parent/child relationships. Dynamic data are stored in form of different logs: container log, observation log, transaction log, tag allocation log (history of all tags allocated). Each log entry has a timestamp. Sun provides us a useful meta schema for static entity. It preserves the completeness of RFID data history. However, it is weak in business process semantics because it does not include different aspects of a business process but location and shipping.

Tuyen Nguyen et al. proposed an event-based data model for EPC Information Service (EPCIS) repository [5]. They also exploit both object-relational and relational strength to make data rich in semantics and build a generalization hierarchy of events to answer queries about historical events recommended by EPCGlobal's EPCIS specification. Their model is based on object-relational approach thus it can handle various kinds of data and relationships efficiently. However, most current RFID application systems depend heavily on relational tables and have difficulty in mapping their Extended Entity Relationship into relational tables.

In this paper, based on DRER model [2] we extend the definition of the physical and conceptual entities in RFID systems and enrich the fields in the tables which are created according to the DRER model. With our extension this data model is rich in semantics and it can support complex RFID queries with temporal constraints such as history, temporal snapshot, or temporal slicing.

The rest of this paper is organized as follows. In section II we firstly present a supply chain application which tracks products with RFID tags at the item level and then we introduce the primary entities and the dynamic interactions between the entities involved in RFID events. In section III we give our model description and express the Dynamic Relationship ER (DRER) model and the tables for DRER model in detail. In section IV we summarize common RFID data tracking and monitoring types, and propose methods to express such queries based on DRER data model. Section V overviews some related work. Section VI concludes our paper and gives some future work.

II. RFID DATA MODELING

A. Example

Consider a large retailer with a supplier and distribution network that tracks objects with RFID tags placed at the item level. Such a retailer sell thousands of items per day through hundreds of stores, and for each such item, it records the complete set of movements between locations, starting at factories in producing countries, going through the transportation network, and finally arriving at a particular store where the item is purchased by a customer. The whole process can be illustrated in Fig. 1.

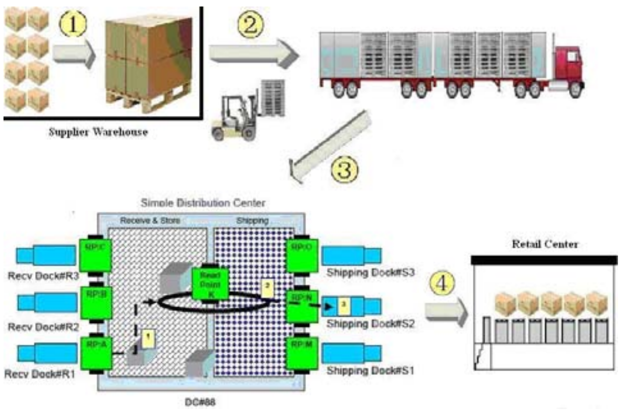

Figure 1. A sample RFID-Enabled Supply Chain System.

Firstly, each EPC-tagged product item is packed into the pallets at the Supplier Warehouse, where the EPC tags of both the cases and the pallets are scanned by an RFID reader R1. Then with the help of forklift the pallets are loaded onto a truck and both the pallets and the truck are scanned by another reader R2. In the following step the truck departs to a Simple Distribution Center. At the Receive Dock#R1, the pallets are loaded and scanned by reader located at Read Point A, and then the pallets enter into the Receive&Store room. When the products order comes, the pallets are placed on conveyor and enter the Shipping room. At the same time, these pallets are scanned by Read Point K. At last, the pallets are scanned by Read Point N and loaded on truck via dock door#S2. Then the truck leaves for a Retail Center where the products will be sold to customers.

In the whole process of the example above, the static data are similar to traditional regular business data and easy to handle by relational databases. Moreover, we also have to maintain the implied relationship between the products experiencing the same business process. When the EPC-tagged products are packed into the pallets and moved from the Warehouse to the Simple Distribution Center, their relationship with entities location and business step change, new relationships between these products and the pallets begin. Obviously, when a change occurs in an EPC-tagged product life, it can include many changes in the relationships of that product with other entities. In other words, the change contains much semantic information that the the data model should well represent. The RFID data model has to consider all of these problems.

B. Fundamental Entities in RFID Systems

In an RFID system, there are two basic categories of data: static data and dynamic data [3][18]. Static data are related to commercial entities and product/service groups, such as location information, product level and serial level information. Dynamic data are specific to individual items. There are two types of dynamic data: instance data such as serial number and date of manufacture, and temporal data such as observations, location and containment changes of objects, which are all captured through EPC-tag readings. Dynamic data grows in quantity as more business is transacted, and refers to things that happen at specific moments in time. Static data does not generally grow merely because more business is transacted (though static data does tend to grow as organizations grow in size), is not typically tied to specific moments in time (though static data may change slowly over time), and provides interpretation for elements of dynamic data. Among all the data, the temporal data are directly related to the fundamental business logic in RFID applications, such as the movement and transaction of products, and are crucial for an RFID data system to track and monitor objects.

There are the following primary physical and conceptual entities that exist in the business processes.

Objects: These include all EPC-tagged objects, such as items, cases, pallets, trucks and so on. For example, the objects in Figure 1 include cases, pallets and trucks.

Readers: RFID readers use radio-frequency signals to communicate with EPC tags and read the EPC values in these tags.

Locations: A location is symbolized to represent where an object is. It can be a warehouse, a retail store, a distribution center, a reader point, or a route between two locations. For example, in Fig. 1 the locations include: supplier warehouse (L001), route from the warehouse to the simple distribution center (L002), retail center (L003) and customers (L004). There are two types of location in simple distribution center in Fig. 1: Read Point (RP) and Business Loaction. The granularity of locations can be defined according to application needs. A location is also associated with an owner.

The distinction between Read Point and Business Location is very much related to the dichotomy between retrospective semantics and prospective semantics discussed above. In general, Read Points play a role in retrospective semantics, while Business Locations are involved in prospective statements. The simple distribution center in Fig. 1 shows a typical use of case consisting of rooms with fixed doorways at the boundaries of the rooms. In such a case, Read Points correspond to the doorways and Business Locations correspond to the rooms. Here the Read Points and Business Locations are not in one-to-one correspondence; the only situation where Read Points and Business Locations could have a 1:1 relationship is the case of a room with a single door.

Business Steps: The business step field of an event specifies the business context of an event: what business process step was taking place that caused the event to be captured? Some common terms such as “shipping” or “receiving” are used to describe business steps.

Disposition: Disposition denotes a business state of an object. Some common terms such as “available for sale” or “in storage” are used to describe disposition. The disposition field of an event specifies the business condition of the event’s objects, subsequent to the event. The disposition is assumed to hold true until another event indicates a change of disposition.

Transactions: There can be business transactions in which EPC is involved. Transaction information may be included in events to record an event’s participation in particular business transactions. For example, a checkout involves a credit card transaction with many EPC readings. Transactions are business-specific, i.e., with different transaction types.

These entities interact with each other to produce corresponding events. Next we will discuss how these entities interact with each other.

C. Dynamic Interactions between RFID Entities

While the above disscussed entities are themselves static in the business process, they dynamically interact with each other and generate movement, workflow, and business logic. These interactions generate events and states changes. The main state changes include:

Object Location Change: For instance, the truck and its loaded pallets leave the warehouse.

Reader Location Change: For instance, a reader is deployed at a new location.

There is also another state change, ownership change, for example, cases unloaded into the retail store. This can be combined with location changes, since a location is always associated with an owner. The information about during which period an object is in certain state is essential and has to be captured.

Business Step Change: In different business processes, when the business context shifts, the business step changes with it.

Disposition Change: At different places, the state of an object changes. For example, the state of an object in storeroom is “in storage” while the state of the object on shelf is “available for sale”.

Besides these state changes, there are also events generated in the interactions, including:

Observations: These are generated when readers interact with objects’ tags (e.g., “EPC 1.1.1 was shipped from storeroom at time 11:39am”).

Aggregation: Aggregation event announces a specific group of EPC-tagged objects were contained in another one (e.g., “at time 17:33, objects of EPCs 1.1.2, 1.1.3 and 1.1.4 were aggregated to the case 1.2.2 at the warehouse”).

Transacted Items: These are generated when an object participates into a transaction (e.g., “at time 09:29am, the cases of EPC 1.1.1, 1.1.2 and 1.1.3 were shipped for purchase order #123”).

All events are associated with timestamps when they occur. The entities have dynamic relationships with each other, which lead to events and states changes.

One of the essential goals of an RFID-enabled application is to track objects and monitor the system at any location, at any time, or both. The data model should be efficient enough for complex queries related to business context and it should be rich in business process semantics so that it could contain different aspects of a business process. This requires an expressive data model that can explicitly represent the history of both events and states, and capture fundamental business logic into the data model itself, therefore complex queries such as tracking and monitoring can be efficitively supported.

In the next section, we introduce the Dynamic Relationship Entity-Relation (DRER) Model that fits perfectly with RFID data.

III. DYNAMIC RELATIONSHIP ER (DRER) MODEL

A. Model Description

DRER is a temporal extension of ER model. In ER model, all entities and relationships are static or current. In an RFID system, entities are static, but all the relationships are dynamic. Here we naturally extend ER model by inheriting all the ER model semantics and simply adding a new relationship – dynamic relationship, as shown in Fig. 2. There are two types of dynamic relationships: relationship that generates events, and relationship that generates state history. Here we use dash lines to represent state-based dynamic relationship, and dash-dot lines to represent event-based dynamic relationship.

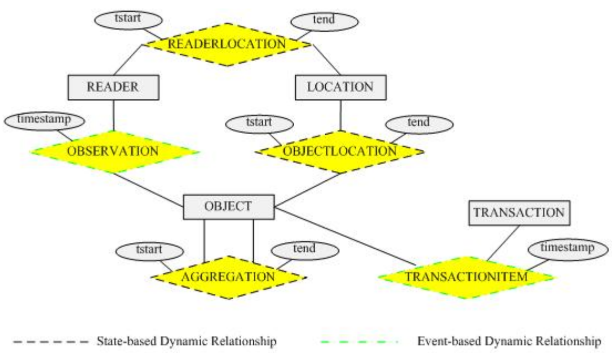

Figure 2. Dynamic Relationship ER model.

Two attributes tstart and tend are associated with a state-based dynamic relationship, to represent the lifespan of that relationship; and an attribute timestamp is associated with an event-based dynamic relationship, to represent the occurring timestamp of the event.

The simplicity of DRER is that, it requires minimum extensions to ER model, by simply adding two types of relationships, each of which is associated with special time-related attributes.

Thus, based on DRER data model, we have following static entities for RFID data (Fig. 2): READER, OBJECT, LOCATION and TRANSACTION. State-based dynamic relationships include: READERLOCATION generated from READER and LOCATION, OBJECTLOCATION generated from OBJECT and LOCATION; Event-based dynamic relationships include: OBSERVATION generated from OBJECT and READER, AGGREGATION generated from OBJECT and itself, and TRANSACTIONITEM generated from OBJECT and TRANSACTION.

It is easy and straightforward to implement DRER model in an RDBMS. There are two types of mappings from the data model to tables. Entities are mapped directly as entity tables. For a state-bsed dynamic relationship between two entities, it is mapped as a table consisting of keys from both entities, plus an interval [tstart, tend] to represent the lifespan in which the relationship or the state exists. For an event based dynamic relationship, it is mapped as a table consisting of keys from both entities, plus a timestamp to represent the time when the event occurs. Normal relationships, if any, can be mapped as tables according to common ER mapping rules.

We use symbol 'UC' to denote now. 'UC' can be represented as 'end-of-time', e.g., "9999-12-31 23:59:59", in the database. The tables for DRER model are described as follows.

B. Entity Tables

READER (reader_epc, name, description)

The READER table records the reader's EPC, name and description of a reader (Table I).

TABLE I. SAMPLE READER

Reader_epc	Name	Description
1.255.1	R1	Packing reader
1.255.2	R2	Departure reader
1.255.3	R3	Unloading reader
1.255.4	R4	Checkout reader

OBJECT (epc, name, description)

The OBJECT table includes the EPC, name and description of an EPC-tagged object (Table II).

TABLE II. SAMPLE OBJECT

Epc	Name	Description
1.1.101	Case	Containing items
1.1.102	Case	Containing items
1.1.103	Case	Containing items
1.1.104	Case	Containing items
1.2.1	Pallet	Containing items
1.3.1	Truck	Loaded with pallets

LOCATION (location_id, name, owner)

The location table defines symbolic business locations used for tracking, including location_id, name, and owner of a location (Table III).

TABLE III. SAMPLE LOCATION

Location_id	Name	Owner
L001	Warehouse A	Supplier A
L002	Route to retail	Carrier B
L003	Retailer store C	Retailer C
L004	Customer	Customer

TRANSACTION (transaction_id, transaction_type)

Since transaction data are business specific, here we simplify a transaction as a record with transaction id and a transaction type (Table IV).

TABLE IV. SAMPLE TRANSACTION

Transaction_id	Transaction_type
TX00001	Retail_checkout
TX00002	Retail_checkout
TX00003	Retail_checkout
TX00004	Retail_checkout

Here we only consider the fundamental attributes. If additional attributes are needed, it can be extended with an extension table. For example, for the READER table, there can be an extension table: READER_EXT (epc, property, value).

C. Dynamic Relationship Tables

OBSERVATION (reader_epc, value, timestamp, bizstep)

This table records the reading data generated from readers, including reader's EPC value, tag's EPC value, the reading timestamp and the steps of business process (Table V).

TABLE V. SAMPLE OBSERVATION

Reader_epc	Value	Timestamp	Bizstep
1.255.1	1.1.1	2010-11-01 10:33:23	Shipping
1.255.1	1.2.1	2010-11-01 10:34:45	Shipping
1.255.2	1.2.1	2010-11-01 10:38:19	Shipping
1.255.3	1.1.1	2010-11-07 14:28:51	Receiving
1.255.4	1.1.1	2010-11-08 17:54:36	Receiving

AGGREGATION (epc, parent_epc, tstart, tend, bizstep)

This table records in which period [tstart, tend] an object (identified by its EPC) is contained in a parent object (identified by its parent EPC) and the steps in business process (Table VI).

TABLE VI. SAMPLE AGGREGATION

Epc	Parent_epc	Tstart	Tend	Bizstep
1.1.1	1.2.1	2010-11-01 10:33:23	2010-11-09 06:22:47	Loading
1.1.2	1.2.1	2010-11-01 10:34:45	2010-11-09 06:22:47	Loading
1.2.1	1.3.1	2010-11-02 07:28:37	2010-11-06 11:39:22	Loading

OBJECTLOCATION (epc, location_id, tstart, tend, disposition)

This table preserves the location history of each object, including an object's EPC, location id, the period [tstart, tend] during which the object stays in that location, and the state of the object (Table VII). (UC denotes 'now' in the examples.)

TABLE VII. SAMPLE OBJECTLOCATION

Epc	Location_id	Tstart	Tend	Disposition
1.1.1	L001	2010-09-30 07:21:39	2010-10-04 07:21:39	In stroage
1.1.1	L002	2010-10-04 07:21:39	2010-10-04 18:26:51	Transported
1.1.1	L003	2010-10-04 18:26:51	2010-10-08 11:37:42	Available for sale
1.1.1	L004	2010-10-08 11:37:42	UC	Sold

READERLOCATION (reader_epc, location_id, position, tstart, tend)

This table keeps the location history of a reader, since a reader can be deployed at different locations. It includes the EPC of a reader, location id, position, and the period during which the reader is located in that location. Here position is a symbolic position of a reader at a location, for example, the loading zone of a warehouse. A location of a reader at certain time can always be found in this relation (Table VIII).

TABLE VIII. SAMPLE READERLOCATION

Reader_epc	Location_id	Position	Tstart	Tend
1.255.1	L001	Packing	2010-12-31 16:24:48	UC
1.255.2	L001	Loading	2010-10-22 19:39:26	UC
1.255.3	L003	Unloading	2010-11-09 14:29:40	UC
1.255.4	L003	Register	2010-11-29 06:22:39	UC

TRANSACTIONITEM (transaction_id, epc, timestamp, disposition)

This table keeps the items in a transaction. It includes a transaction id, EPC of the object in the transaction, the timestamp when the object's transaction occurs, and the disposition of the object (Table IX).

TABLE IX. SAMPLE TRANSACTIONITEM

Transaction_id	Epc	Timstamp	Disposition
TX00001	1.1.1	2010-12-22 18:26:29	Sold
TX00001	1.1.2	2010-12-22 18:26:29	Sold
TX00001	1.1.3	2010-12-22 18:26:29	Sold

IV. TRACKING AND MONITORING RFID DATA

A significant benefit of the temporal modeling is the power to support complex RFID queries. Most RFID queries are temporal queries with constraints such as history, temporal snapshot, or temporal slicing. There are more complex ones such as temporal joins and temporal aggregates. We summarize these queries as three main categories: RFID Object Tracking and RFID Reader Observation Tracking and RFID Object Monitoring. RFID Object Tracking is to track RFID objects including missing objcts. RFID Reader Observation Tracking aims at monitoring the readers and objects involved in reading process queries. RFID Object Monitoring is to monitor the states of RFID objects and the RFID system. Next we summarize common RFID data tracking and monitoring types, and propose methods to express such queries based on DRER data.

A. Methods for RFID Data Tracking

RFID Object Tracking tracks the change history of an object's states and detects some missing objects.

RFID Object Tracking

Q1. Find the location history and the corresponding disposition of an object with EPC value 'EPC'.

```
SELECT *
FROM OBJECTLOCATION
WHERE epc = 'EPC'
```

Q2. Find the disposition of an object with EPC value 'EPC' at time 'T'.

```
SELECT disposition
FROM OBJECTLOCATION
WHERE tstart<='T'
        AND tend>='T'
        AND epc = 'EPC'
```

Q3. Find the current owner of an object with EPC value 'EPC'.

```
SELECT owner
FROM LOCATION
WHERE location_id =
        SELECT location_id
        FROM OBJECTLOCATION
        WHERE epc = 'EPC'
```

Q4. Find the location name of an object with EPC value 'EPC' at time T.

```
SELECT name
FROM LOCATION
WHERE location_id =
        SELECT location_id
        FROM OBJECTLOCATION
        WHERE epc= 'EPC'
              AND tstart<='T'
              AND tend>='T'
```

Missing RFID Object Detection

There are two scenarios for Missing RFID Object Detection. The first one is Missing RFID Object Tracking, to locate when and where an object was lost, knowing the lost object's EPC. This means that the object appeared at previous location, but not at current location.

Q5. Find when and where object 'MEPC' was lost and the dispositon of the object.

```
SELECT location_id, tstart, tend, disposition
FROM OBJECTLOCATION
WHERE epc = 'MEPC'
        AND tstart = (SELECT MAX (o.tstart)
                      FROM OBJECTLOCATION o
                      WHERE o.epc = 'MEPC')
```

The second scenario is Possible Missing RFID Object Searching, to search if there is any missing object at a certain location C, knowing that at a previous location L and timestamp T, all objects were complete. This can be done by comparing the two sets of objects between location C and location L.

Q6. Check if there are missing objects at current location C, knowing that all objects were complete at previous location L at time T.

```
SELECT l.epc FROM OBJECTLOCATION l
WHERE l.location_id = 'L'
        AND l.tstart <= 'T'
        AND l.tend >= 'T'
        AND l.epc NOT IN (
                SELECT c.epc
                FROM OBJECTLOCATION c
                WHERE c.location_id = 'C')
```

RFID Object Identification

Since every RFID object is uniquely identified by its EPC, it is easy to identify an object:

Q7. A customer returns a product with EPC 'XEPC'. Check if this product was sold from this store (location 'L003')

```
SELECT *
FROM OBJECTLOCATION
WHERE epc = 'XEPC'
        AND location_id = 'L003'
```

RFID Object Moving Time Inquiry

One common query is to find how long it takes for an object to move from one location to another.

Q8. How long did it take to supply object 'OEPC' from location S to location E?

```
SELECT (e.tstart – s.tstart) AS supplying_time
FROM OBJECTLOCATION e, OBJECTLOCATION s
WHERE e.epc = 'OEPC'
        AND s.epc = 'OEPC'
        AND s.location_id = 'S'
        AND e.location_id = 'E'
```

B. Methods for Reader Observation query

Reader observation queries express the scenarios of the reading process and the conditions of the readpoints.

Q9. Find the readers read 'EPC' over time range (T1, T2)

```
SELECT reader_epc
FROM OBSERVATION
WHERE value= 'EPC'
        AND timestamp > T1
        AND timestamp < T2
```

Q10. Find the objects' EPCs which were seen by reader R1 over time range (T1, T2)

```
SELECT value
FROM OBSERVATION
WHERE reader_epc = R1
        AND timestamp > T1
        AND timestamp < T2
```

C. Methods for RFID Data Monitoring

RFID Object Monitoring is to monitor the states of RFID objects and the RFID system. These include snapshot inquiry, temporal slicing inquiry, temporal join query, temporal quantity, and aggregation examination.

RFID Object Snapshot Query

By specifying a snapshot timestamp, it is easy to monitor the snapshot information of any RFID objects, including snapshot locations, containment, observations or transactions.

Q11. Find the direct container of object 'EPC' at time T.

```
SELECT parent_epc
FROM AGGREGATION
WHERE epc = 'EPC'
        AND tstart <= 'T'
        AND tend >= 'T'
```

RFID Object Temporal Slicing Query

This query retrieves object information during a temporal interval.

Q12. Find items sold to customers in the last hour.

```
SELECT epc FROM OBJECTLOCATION
WHERE locaion_id = 'L04'
        AND tend = 'UC'
        AND tstart <= sysdate – (1/24)
```

RFID Object Temporal Join Query

Temporal join query will retrieve information by joining multiple relations on certain temporal constraint.

Q13. This case (with epc 'TEPC') of meat is tainted. What other cases have ever been put in the same pallet it?

```
SELECT a2.epc
FROM AGGREGATION a1, AGGREGATION a2
WHERE a1.parent_epc = a2.parent_epc
        AND a1.epc = 'TEPC'
        AND overlaps (a1.tstart, a1.tend, a2.tstart,
                a2.tend)
```

Where overlaps ( ) is a user-defined scalar function to check if two intervals overlap. User-defined scalar temporal functions can be defined to simplify temporal queries.

Temporal Quantity of RFID Data

This query will summarize quantity information at certain snapshot or interval.

Q14. Find how many items loaded into the store 'L003' on 11/09/2010

```
SELECT count (epc)
FROM OBJECTLOCATION
WHERE location_id = 'L003'
        AND tstart <= '2010-11-09 00:00:00'
        AND tend >= '2010-11-09 00:00:00'
```

RFID Object AGGREGATION Queries

RFID aggregation queries are queries that retrieve the containment relationships between RFID objects. These queries are normally interleaved with other temporal RFID queries. Two special cases are recursive containment queries: RFID Obect Sibling Search: find all the sibling objects of a container object, and RFID Object Ancestor Search: find all the ancestor container objects of an object. The following shows an example of sibling search (ancestor search can be done similarly by swithching parent and child attributes).

Q15. RFID Sibling Object Search. Find all objects contained in object 'PEPC'.

```
WITH RECURSIVE all_sub (parentepc, epc) AS
(
SELECT parentepc, epc
 FROM AGGREGATION
 WHERE parentepc = 'PEPC'
 UNION
 SELECT a.parentepc, c.epc
 FROM all_sub a, AGGREGATION c
 WHERE a.epc = c.parentepc
)
SELECT *
```

Based on the DRER schema, we can define other queries which are related to the specific business and we can also specify constraints and business intelligence queries, such as automatic shipping notice, low inventory alert, and trend analysis.

V. RELATED WORK

RFID technology makes a fast progress in recent years and promises a bright future of labor cost reduction, business process automation and inventory inaccuracy reduction, etc. However, there are many challenges for data management we need to consider when deploying this technology [6][7][8]. Many researchers devotes their time to find solutions for these problems including building warehouse model [9][10], cleansing anomalies in RFID reads [11], using bitmap datatype for representing collections of EPCs [12], extracting valuable information from RFID data for supply chain management [22] and efficient data interpretation and compression over RFID streams [23].

For RFID data, an observational result is created dynamically and the location and inclusion relation of an object changes with time. Thus, RFID related data [9][19]should be expressed as an adequate data model at the stage of the related application, including tracking [20][21], monitoring and application integration. Besides, event processing systems is another area of research. RFID systems generate a stream of raw events, which themselves can form higher level events. Work in this area has focused on efficient processing of very large data streams [13][14], RFID simple and complex event processing [24], policy-driven RFID event management using an XML-based policy definition language [25], definition of complex events, extensions to standard query languages to account for unique temporal characteristics of RFID events [15][16], and [14] which cope with noise in the RFID stream by defining probabilistic events over low-level events.

VI. CONCLUSIONS AND FUTURE WORK

RFID technology holds the promise of real-time identifying, locating, tracking and monitoring physical objects, and can be used for a wide range of applications. To achieve these goals, RFID data have to be collected and expressively modeled. RFID data, however, have their own unique characteristics–including aggregation, location, temporal and history-oriented–which have to be fully considered and integrated into the data model. In this paper, we introduce the fundamental characteristics of RFID data. We also explore the physical and conceptual entities in the RFID application systems as well as the dynamic interactions between these entities. Then based on the Dynamic Relationship ER model, we extend the definition of the fundamental entities in RFID systems and enrich the fields in the tables which are created according to the DRER model. With our extension this data model is rich in semantics and it can support complex RFID queries with temporal constraints such as history, temporal snapshot, or temporal slicing. Lastly, we we summarize common RFID data tracking and monitoring types, and propose methods to express such queries based on DRER data. We can see that much more semantics information can be displayed in the process of the SQL queries.

In a networked environment of RFID readers, enormous data is generated from proliferation of RFID readers. Accordingly, in RFID environment, the database becomes more pervasive and various data quality issues regarding data legacy, data uniformity and data duplication arise. The raw data generated from the readers can't be directly used by the application. There are various data processing and management problems such as missed readings and duplicate readings which hinder wide scale adoption of RFID systems [26]. In the future, our work will focus on detecting and correcting data recording errors to increase data quality, including the correct reading identification and redundant data filtering.

ACKNOWLEDGMENT

This work was supported by the Fundamental Research Funds for the Central Universities.

REFERENCES

[1] Hector Gonzalez et al., "Modeling massive RFID Data Sets A Gateway-Based Movement Graph Approach", IEEE transactions on knowledge and data engineering, vol. 22, no. 1, pp. 90-103, January 2010.

[2] Fusheng Wang and Peiya Liu, "Temporal Management of RFID Data", in Proc. of VLDB, 2005

[3] Mark Harrison, "EPC information service – data model and queries", White paper, Auto-ID Center for Manufacturing, University of Cambridge, February 2003.

[4] "Sun java system RFID software", http://sun.com/rfid, 2009.

[5] Tuyen Nguyen, Young-Koo Lee, Rezwanul Huq, Byeong-Soo Jeong, and Sungyoung Lee, "A Data Model for EPC Information Services", in DEWS'2007, March 2007.

[6] Venkat Krishnamurthy et al, "Managing RFID data", in proceedings of the 30th VLDB conference, Toronto, Canada, 2004.

[7] Jun Rao, Sanqeeta Doraiswamy, Hetal Thakkar and Latha S. Colby, "A Defferred Cleansing Method for RFID Data Analytics", in proceedings of the 32nd VLDB Conference, pp. 175-186, 2006.

[8] Yu-Ju Tu, Selwyn Piramuthu, "A Decision Support Model for Filtering RFID Read Data", IEEE ADCOM 2008.

[9] Hector Gonzalez, Jiawei Han, Xiaolei Li, Diego Klabjan. "Warehousing and Analyzing Massive RFID Data Sets", in proceedings of the 22th International Conference on Data Engineering, page 83, 2006.

[10] Hector Gonzalez, Jiawei Han, and Xiaolei Li, “Mining compressed commodity workflows from massive rfid data sets”, in proceedings of CIKM, November 2006.
[11] Shawn R. Jeffery, Minos Garofalakis, and Michael J. Franklin, “Adaptive cleaning for RFID data streams”, in proceedings of VLDB, 2006.
[12] Ying Hu, Seema Sundara, Timothy Chorma, Jaqannathan Srinivasan, “Supporting RFID-based item tracking applications in Oracle DBMS using a bitmap datatype”, in proceedings of VLDB, 2005.
[13] Nodira Khoussainova, Magdalena Balazinska, and Dan Suciu, “Peex: Extracting Probabilistic Events from RFID Data”, in proceedings of 2008 International Conference on Data Engineering, pp. 1480-1482, 2008.
[14] Chun-Hee Lee and Chin-Wan Chung, “Efficient Storage Scheme and Query Processing for Supply Chain Management Using RFID”, in proceedings of SIGMOD 2008, pp.291-302, 2008.
[15] Yijian Bai, Fusheng Wang, Peiya Liu, C. Zaniolo, and Shaorong Liu, “RFID Data Processing with a Data Stream Query Language”, in proceedings of ICDE, pp. 1184-1193, 2007.
[16] Euqene Wu, Yanlei Diao, and Shariq Rizvi, “High-Performance Complex Event Processing Over Streams”, in proceedings of SIGMOD’06, pp. 407-418, 2006.
[17] H. Gregersen and C. S. Jensen. Temporal Entitiy-Relationship Models – a Survey. IEEE transactions on Knowledge and Data Engineering, 11(3):464-497, 1999.
[18] The EPCglobal Network and the Global Data Synchronization Network (GDSN): Understanding the Information & the Information Networks, http://www.epcglobalinc.org/news/position_papers.html, October 2004.
[19] Jiawei Han, Hector Gonzalez, Xiaolei Li, Diego Klabjan. “Warehousing and Mining Massive RFID Data Sets”, in proceedings of 2nd International Conference on Advanced Data Mining and Applications, pp.1-18 2006.
[20] Ying Hu, Seema Sundara, Timothy Chorma, Jagannathan Srinivasan. Supporting RFID-based Item Tracking Applications in Oracle DBMS Using a Bitmap Datatype. In proceedings of the 31st International Conference on Very Large Databases, pp.1140-1151, 2005.
[21] Elio Masciari. RFID data management for effective objects tracking.Proceedings of the 2007 ACM symposium on Applied computing, pp.457-461, 2007.
[22] Chun-Hee Lee, Chin-Wan Chung, “RFID Data Processing in Supply Chain Management Using a Path Encoding Scheme”, IEEE Transactions on Knowledge and Data Engineering, vol.23, pp.742-758, May 2011.
[23] Nie Yanming, Cocci Richard, Diao Yanlei, Prashant Shenoy, “SPIRE: Efficient Data Interpretation and Compression over RFID Streams,” IEEE Transactions on Knoledge and Data Engineering, vol.21, pp 521-535, March 2011.
[24] Chunkai Zhang, Cong Li, “Rule-Matching based Method for RFID Event Processing”, in proceedings of 2010 4th International Conference on Genetic and Evolutionary Computing, pp 272-275,2010.
[25] Hyun Ahn, Bumjo Park et al, “RXPDL: An XML-based Policy Definition Language for Policy-driven RFID Event Management Agents”, in proceedings of 2011 13th International Conference on Advanced Communication Technology, pp.71-75, 2011.
[26] Yu-Ju Tu, Piramuthu S., “A Decision-Support Model for Filtering RFID Read Data in Supply Chains”, IEEE Transactions on Systems, Man, and Cybernetics Part C: Applications and Reviews, vol.41, pp 268-273, March 2011.

Kun Bai is a doctoral student in School of Navigation at Dalian Maritime University. He received his BS in School of Information from Dalian Maritime University. His primary research interests lie in traffic information system and the internet of things.

Shaoxi Li is an instructor in School of Navigation at Dalian Maritime University. Now he is a doctoral student in School of Navigation at Dalian Maritime University. He received his MS degree in School of Navigation from Dalian Maritime University. His primary research interests include electronic navigation chart and smart navigation.

Tao Liu is a doctoral student in the School of Navigation at Dalian Maritime University. He received his MS degree in Dalian Maritime University. His major research interests include geographic information system and graph and image processing.

Jing Tian is an assistant instructor in Liaoning Police Academy. She received her MS degree in Dalian Maritime University. Her major research interests include multimedia technology.